

NWG/RFC# 555

Response to Critiques of the Proposed MP

JEW 30-JUL-73 12:12 1799

RFC 555 / NIC 17993

RECEIVED

RFC # 555

NIC # 17993

AUG 1 1973

James E. White (JEW)

SRI-ARC

27 July 73

Response to Critiques of the Proposed Mail Protocol

Discussion

A number of people have responded to my proposal for a Mail Protocol (JEW RFC 524 -- 17140,2:y). In the current RFC, I've attempted to collect and respond to the questions, complaints, and suggestions that various individuals in the Network community have offered. I intend to critique myself in a forthcoming RFC.

I hope that dialog on the protocol proposal will continue, and that others will join in the discussion. I will respond via RFC to any additional critiques I receive (I hope there'll be many).

With such support available to it, the MP server process can take on the role of an end node process that can act as a Mailbox Manager (M) within the user-server interchange within the constraints of the protocol. The server simply prompts for the ID, gets it, opens a connection to the User Verification Server, presents to it the individual's name and requested ID, receives a positive or negative response, and deals with the original user process accordingly.

Example:

Suppose a user process opens a connection to UCLA-MP's server process, invokes the Delivery function, and in the course of the interchange identifies the author as Robert at USC-100.

The implementation of UCLA-MP's server process then prompts for proof. In all Delivery transactions, the user is the author in who he claims to be. It therefore prompts for an ID in response to the request. From the user process, and receiver is returned the response 'ID approved OK'.

UCLA-MP's server then connects to the USC's server, invokes the User Verification function there, specifying 'FROM:Robert at usc-100 (C4)' and

I. QUESTIONS

HOW DOES THE SERVER VERIFY AN ID?

References:

(DHC JBP RFC 539 -- 17644,3g:gy)

Discussion:

One postulates the existence of AT LEAST ONE host whose Mail server process implements the User Verification Function (JEW RFC 524 -- 17140,5f7:gy). Any process can contact that server, give him the name of any Individual in the Net and a test Id, and the server will determine whether or not the Individual and Id agree.

The NIC, for one, will without question provide this service.

With such support available to it, ANY FTP server process can then require (of any or all user processes that contact it) an ID command wherever it wishes within the user-server interchange (within the constraints of the Protocol). The server simply prompts for the Id, gets it, opens a connection to the User Verification Agent, presents to it the Individual's name and purported Id, receives a positive or negative response, and deals with the original user process accordingly.

Example:

Suppose a user process opens a connection to UCLA-NMC's server process, invokes the Delivery function, and in the course of the interchange identifies the Author as Roberts at USC-ISI.

The implementors of UCLA-NMC's server process chose to require proof, in all Delivery transactions, that the Author is who he claims he is. It therefore prompts for an Id in response to the AUTHOR command from the user process, and receives in return the command 'ID arpawheel <CA>'.

UCLA-NMC's server then connects to the NIC's server, invokes the User Verification function there, specifying 'REQUESTOR roberts @ usc-isi <CA>' and

'ID arpawheel <CA>'. The NIC informs UCLA-NMC that the Id is incorrect.

4a2c3

ACF, PRO UCLA-NMC then rejects the original ID command.

4a2c4

Of course, the Protocol does not require that a server demand Ids from users that contact it. Servers who choose not to require proof of identity simply never prompt for ID commands, and treat any they receive as NOPs. For such implementations (which represent the current, FTP mail protocol situation), no third-part interchanges are ever required.

4a2d

Each user in the Net has a single Id that he uses throughout the Net for purposes of sending and receiving mail. That Id need not (but may, either coincidentally or by design) have any other use. In particular, a user's Id is independent of the passwords by which he gains access to accounts that he might possess on hosts around the Net.

4a2e

Of course, a user could and might see to it that his passwords and Id are the same. The NIC, for example, might require that a user log in to its system with NIC ident and Id, rather than with host name and password, as it does currently.

4a2e1

I emphasize again that Ids have nothing whatsoever to do with accounting. UCLA-NMC doesn't force the Author to prove his identity so UCLA has someone to whom it can bill the resources consumed in processing the Delivery transaction. It does so to prevent Jim White from authoring a piece of mail and claiming that Larry Roberts wrote it.

4a2f

UCLA-NMC does have the option of requiring that a user process log in before it delivers mail so that it can be billed for the resources it uses. The appropriate commands to require of the user process are USER, PASS, and ACCT. But, the billing process is separable from that of identifying Author, Clerk, etc.

4a2f1

The NIC, for example, in its role as a Distribution Agent, might establish an account at UCLA-NMC to use whenever it delivers mail there. UCLA-NMC will bill ALL of the NIC's activity at UCLA to that account. But when the NIC delivers a piece of mail it claims

II. ~~EXAMPLE~~ was authored by Larry Roberts, UCLA-NMC may still wish to verify that claim. Hence the ID command.

ACK, PROGRESS REPORT, OR REPLY WITH NO REFERENCE SERIAL NUMBER

References: 17804,)

(DHC JBP RFC 539 -- 17644,3h:gy)

Discussion: 0 objection to defining the terminating

A Delivery of type POSITIVE or NEGATIVE ACKNOWLEDGMENT, PROGRESS REPORT, or REPLY requires a Reference Serial Number of the user process. Should the server determine that one is lacking when the final EXIT command is given, he should reject the EXIT command with an appropriate error response.

The same applies in the Distribution function: a Reference Serial Number MUST be specified if the Delivery Type is REPLY. Level Protocol, I don't consider that as a protocol for direct use by users. The Protocol document is deficient in that it doesn't state the above.

The concept I want to extract from the DUEP and exploit is the idea of a single process with many sub-processes, not the philosophy of a structured-standard command language for use by users where (the latter may be a good idea, too, but it's not the one I'm concerned with at the moment).

I don't think that designing a protocol to govern an exchange between processes is the same thing as designing a protocol to mediate a conversation between a process and a human user. Using ASCII commands suggests (as it did for PDP, V37, etc.) that the latter problem is the one being addressed; it's not.

USING TELNET TO ABORT TO TERMINATE CERTAIN ECNAMES

References:

(1st -- 17804,)

(2nd -- 17822, 1st:01)

II. COMPLAINTS

TERMINATING BOTH THE SUBSYSTEM AND FUNCTIONS WITH EXIT

References:

(AAM -- 17404,)

Discussion:

I have no objection to defining two terminating commands, one to exit a function, the other to exit the subsystem. I guess I'd suggest defining a command 'GO <CA>' to be used to terminate a function.

I don't believe, however, that's it's necessary to distinguish the two cases to avoid confusion by human users.

Even though the command language is ASCII, rather than binary, and even though I've adopted Mike Padlipsky's concept of a Unified USER Level Protocol', I don't consider that MP is a protocol for direct use by humans (although nothing can STOP a human user from speaking MP if he has access to a TELNET user program and is determined to do so).

The concept I mean to extract from the UULP and exploit is its model of a single process with many subsystems, not its philosophy of a Network-standard command language for use by human users (the latter may be a good idea, too, but it's not the one I'm concerned with at the moment).

I don't think that designing a protocol to govern an exchange between processes is the same task as designing a protocol to mediate a conversation between a process and a human user. Using ASCII commands suggests (as it did for FTP, RJE, etc.) that the latter problem is the one being addressed; it's not.

USING TELNET GO AHEAD TO TERMINATE CERTAIN CCMMANDS

References:

(AAM -- 17404,)

(BCC -- 17822, 1a:gy)

11. COMPLAINTS

TERMINATING BOTH THE SUBSYSTEM AND FUNCTION WITH EXIT

References:

(AAM -- 17404)

Discussion:

I have no objection to defining two terminating commands, one to exit a function, the other to exit the subsystem. I guess I'd suggest defining a command 'END' to be used to terminate a function.

I don't believe, however, that it's necessary to distinguish the two cases to avoid confusion by human users.

Even though the command language is ASCII, rather than binary, and even though I've adapted Mike Padilsky's concept of a Unified User Level Protocol, I don't consider that it is a protocol for direct use by humans (although nothing can STOP a human user from speaking). If he has access to a TRIST user program and is determined to do so.

The concept I mean to extract from the UULP and exploit is its lack of a single process with any subsystem, not its philosophy of a network-standard command language for use by human users (the latter may be a good idea, too, but it's not the one I'm concerned with at the moment).

I don't think that designing a protocol to govern an exchange between processes is the same task as designing a protocol to mediate a conversation between a process and a human user. Using ASCII commands suggests (as it did for FTP, QJL, etc.) that the latter problem is the one being addressed; it's not.

USING TRIST GO AHEAD TO TERMINATE CERTAIN COMMANDS

References:

(AAM -- 17404)

(RTE -- 17822, 17971)

(DHC JBP RFC 539 -- 17644,3b:gy) 5b1c

Discussion: 5b2

Agreed. My mistake. 5b2a

I simply have a strong distaste for the current FTP convention of terminating commands whose argument may itself contain CR LF with 'CR LF . CR LF'. That seems a little extravagant to me. Personally, I'd prefer a single NVT character as a delimiter. 5b2b

<CA2> only terminates two MP commands (COMMENTS and TEXT). Some NVT character (ESC? ETX? ...) can easily be chosen that need not appear (and can therefore be prohibited from appearing by the Protocol) in the argument to either of those commands. 5b2c

SUBSYSTEM OR SEPARATE, RJE-LIKE SERVER PROCESS 5c

References: 5c1

(DHC JBP RFC 539 -- 17644,4a:gy) 5c1a

(AAM -- 17404,) 5c1b

(ADO RFC 552 -- 17809,3:y) 5c1c

Discussion: 5c2

There are two separable issues here: 5c2a

(1) Server Process Proliferation or Not? 5c2a1

If the consensus of the Network community is that Padlipsky's UULP approach to protocol design and implementation is in fact superior to the current scheme, which calls for the implementation of each new Network protocol as a distinct server process with its own contact socket, then we should begin to embrace that concept and begin reshuffling existing protocol implementations accordingly. Even more surely, NEW protocols (like MP), should be designed in accordance with the new standards, not the old. 5c2a1a

I think Buz Owen's suggestion (ADO RFC 552 -- 17809,3:y) -- that a skeletal UULP be defined, a socket assigned to server processes which

(DNC 180 252 -- 17800, 252)

Discussion:

Agreed. My mistake.

I simply have a strong dislike for the current
convention of terminating commands whose argument
itself contains a LF with 'LF'. That seems
a little extravagant to me. Personally, I'd prefer a
single NVT character as a delimiter.

<C2> only terminates two commands (COMMENT and
TEXT). Some NVT character (ESC? ETC? ...) can easily
be chosen that need not appear (and can therefore be
prohibited from appearing by the protocol) in the
argument to either of those commands.

SYSTEMS OF SEPARATE, B2-LIKE SERVER PROCESS

References:

(DNC 180 252 -- 17800, 252)

(AAS -- 17400)

(AAS 252 -- 17800, 252)

Discussion:

There are two separate issues here:

(1) Server Process Termination at NVT

If the consensus of the network community is that
pathology is a valid approach to protocol design and
implementation is in fact superior to the current
scheme, which calls for the implementation of
each new network protocol as a distinct server
process with its own contact socket, then we
should begin to embrace that concept and begin
rebuilding existing protocol implementations
accordingly. Even more surely, new protocols
(like SP), should be designed in accordance with
the new standards, not the old.

I find that Owen's suggestion (AAS 252 --
17800, 252) -- that a skeleton NVT be defined, a
socket assigned to server processes which

implement it, and MP defined as a subsystem under it -- is excellent. I retract my suggestion (JEW RFC 524 -- 17140,3a2:gy) in favor of Owen's.

5c2a1b

I further suggest that the latest revision of FTP (NJN RFC 542 -- 17759,) be similarly implemented (i.e., as a UULP subsystem), rather than implemented temporarily under a new socket and later moved over to socket 3 as suggested in RFC 542.

5c2a1c

(2) RJE's Model for FTP Use or Not?

5c2a2

If both MP (as currently defined) and RJE were instated as UULP subsystems, they would still embrace different philosophies regarding their use of FTP. As the person who proposed and fought for the current RJE model (i.e., to its use of FTP), I (still) believe it to be an elegant one, more elegant by far than the one I've proposed for MP.

5c2a2a

An alternative I considered and discarded SOLELY for reasons of efficiency (neglecting, perhaps, the issue of cleanness of implementation), is that the command currently defined as 'FILE <CA>' (JEW RFC 524 -- 17140,4g2a:gy), both in specifying Content and in the Citation Retrieval function, be 'FILE <fileaddr> <CA>' instead.

5c2a2b

The server is then obliged to retrieve the Content of the Mail from the designated server process via a third-party exchange.

5c2a2b1

The redefined FILE command would be similar to the LOCATION command, except that the former would specify JUST Content (and none of the other Static Attributes), and that the Server must retrieve the file (which may be a temporary file created by the user process) in real time, i.e. BEFORE it sends its response to the FILE command.

5c2a2c

This alternative eliminates the need to borrow the BYTE, SOCK, PASV, TYPE, STRU, MODE, REST, and SITE commands from FTP (JEW RFC 524 -- 17140,7c1:gy). It also allows the user process the flexibility of specifying a file at a host other than his own.

5c2a2d

implemented it, and WP defined as a subsystem under
it -- is excellent. I retract my suggestion (JUL
PFC 224 -- 17100, 17101) in favor of WP's.

I further suggest that the latest revision of WP
(JUL PFC 223 -- 17100) be similarly implemented
(i.e., as a ULP subsystem), rather than
implemented temporarily under a new socket and
later moved over to socket 3 as suggested in PFC
223.

(2) WP's Model for FTP Use of Host

If both WP (as currently defined) and PFC were
implemented as ULP subsystems, they would still
exhibit different philosophies regarding their
use of FTP. As the person who proposed and
fought for the current PFC model (i.e., to its
use of FTP), I (still) believe it to be an
elegant one, more elegant by far than the one
I've proposed for WP.

An alternative I considered and discarded early
for reasons of efficiency (routing, perhaps,
the issue of cleanliness of implementation), is
that the command currently defined as 'FILE <C>' (JUL
PFC 224 -- 17100, 17101), both as
specifying Content and as the Citation Retrieval
function, be 'FILE <fileid> <C>' instead.

The server is then obliged to retrieve the
Content of the file from the designated server
process via a third-party exchange.

The redefined FILE command would be similar to
the LOCATION command, except that the former
would specify JUST Content (and none of the other
Static Attributes), and that the server end
retrieve the file (which may be a temporary file
created by the user process) in real time, i.e.,
before it sends the response to the FILE command.

This alternative eliminates the need to borrow
the FILE, SOCK, PASS, TYPE, MODE, REST, and
SITE commands from FTP (JUL PFC 224 --
17100, 17101). It also allows the user process
the flexibility of specifying a file at a host
other than his own.

III. SUGGESTION After some thought, I think I agree with Crocker and Postel that theirs is the better implementation.

5c2a2e

DISCUSSION: As they point out, however, this implementation introduces the problem of somehow reconciling the desire to permit (in general) the transfer of mail files without requiring a login, with a server's inability to distinguish that case from the general case of file retrieval (for which many hosts will require a login).

5c2a2e1

USE OF THE DATE FORM 1/2/73 (JAN 2 OR FEB 1?)

5d

References:

5d1

(RCC -- 17822, 1b)

5d1a

Discussion: As for our forwarding leave upon the Protocol Agreed.

5d2

5d2a

ORDER OF PARAMETER SPECIFICATION

5e

References:

5e1

(DHC JBP RFC 539 -- 17644, 31:gy)

5e1a

Discussion:

5e2

The Protocol does not, as Crocker and Postel state, impose an order upon command specification within a function (see for example, JEW RFC 524 -- 17140, 5f1b:gy).

5e2a

Having considered their suggestion only briefly, it does seem to me appropriate to impose some constraints on the order of parameter specification by the user. Off hand, the order suggested -- Dynamic, Optional, Static -- seems good.

5e2b

After some thought, I think I agree with Crocker
and Postal that this is the better
implementation.

As they point out, however, this
implementation introduces the problem of
somehow reconciling the desire to permit (in
general) the transfer of mail files without
requiring a login, with a server's inability
to distinguish that case from the general case
of file retrieval (for which many hosts will
require a login).

USE OF THE DATE FORM 1/1/73 (JAN 1 1973)

Reference:
(PCC -- 17822, 1b)
Discussion:
Agreed.

ORDER OF PARAMETER SPECIFICATION

Reference:
(PCC 17822, 1b) -- 17822, 1b)

Discussion:

The protocol does not, as Crocker and Postal state,
impose an order upon certain specification within a
function (see for example, the PCC 17822 --
17822, 1b).

Having considered their suggestion only briefly, it
does seem to me appropriate to impose some constraints
on the order of parameter specification by the user.
Of course, the order suggested -- Dynamic, Optional,
Static -- seems good.

III. SUGGESTED ADDITIONS

FORWARDING AT DELIVERY TIME

References:

(DHC JBP RFC 539 -- 17644,4b:g)

Discussion:

Including provision for the forwarding of mail at Delivery Time, in contrast to sometime after Delivery in response to a specific Forward request (i.e., function), seems to me a useful addition to the Protocol.

As Crocker and Postel note, only one of the three mechanisms for such forwarding bears upon the Protocol (although the Protocol might mention the other two and either encourage or discourage their use).

I suggest the following reply format, however, rather than the one suggested by Crocker and Postel (DHC JBP RFC 539 -- 17644,4b3c2:gy):

476 <localname> -- is his location.

DEFAULT SIGNATURE SHOULD BE THE AUTHOR

References:

(DHC JBP RFC 539 -- 17644,3c:gy)

Discussion:

Agreed.

LEVELS OF INTERRUPT

References:

(DHC JBP RFC 539 -- 17644,3d:gy)

Discussion:

I see no value to defining numeric shades of urgency, unless the Protocol suggests some particular action the server might take in response to each one.

The whole notion of flagging some pieces of mail as urgent seems to me useless unless the MP server process (not the human recipient) takes some kind of special

III. SUGGESTED ADDITIONS

FORWARDING AT DELIVERY TIME

Reference:

(RVC 252 RVC 252 -- 17931, 18:15)

Discussion:

Including provision for the forwarding of mail at delivery time, in contrast to sometime after delivery in response to a specific forward request (i.e., function), seems to me a useful addition to the Protocol.

As Crocker and Postal note, only one of the three mechanisms for such forwarding bears upon the Protocol (although the Protocol might mention the other two and either encourage or discourage their use).

I suggest the following reply format, however, rather than the one suggested by Crocker and Postal (RVC 252 RVC 252 -- 17931, 18:15):

<localname> -- is his location.

DEFAULT SIGNATURE SHOULD BE THE AUTHOR

Reference:

(RVC 252 RVC 252 -- 17931, 18:15)

Discussion:

Agreed.

LEVELS OF INTEREST

Reference:

(RVC 252 RVC 252 -- 17931, 18:15)

Discussion:

I see no value in defining numeric shades of interest, unless the Protocol suggests some particular action the server might take in response to each one.

The whole notion of flagging some pieces of mail as urgent seems to me useless unless the server process (not the human recipient) takes some kind of special

19. action for urgent mail, BEFORE the human recipient would otherwise be apt to read the mail. If one accepts that argument, there's clearly no point to defining shades of urgency if they have meaning only to the human recipient. True, any pair of human users could privately agree on meanings, but it seems to me preferable to define those meanings formally or not at all.

WARNING THE SERVER OF THE SIZE OF MAIL

References:

(DHC JBP RFC 539 -- 17644,3f:gy)

Discussion:

Agreed. Further suggestions as to the implementation?

DISCOURAGING SERVERS FROM REQUIRING LOGINS

References:

(DHC JBP RFC 539 -- 17644,3j:gy)

Discussion:

Agreed. This is not a new issue.

IV. META-COMMENTS

SIZE OF THE PROTOCOL DOCUMENT

References:

(RCC -- 17822,1e:gy)

Discussion:

I offer an apology for the format of the the Protocol document. It differs radically from that of previous protocol documents (e.g., FTP, RJE), and is certainly not tutorial in its orientation. The glossary is a device I found useful in designing the Protocol. If the substance of the Protocol were agreed upon, then friendlier documentation would have to be written. The choice of approach was greatly affected by my own time constraints.

As I find time, I would like to define the minimum implementation subsets that Clements requests. For the moment, consider the command breakdown below. It represents the case where the server permits only the function by which mail is delivered to users in his host. It has the following attributes:

(1) It supports all of the functions of the current FTP mail protocol. In addition,

(2) It makes specification of author and title explicit, avoiding the current problem of multiple headers (one supplied by the server, the other embedded by the user in the text of the message),

(3) It allows the text of the message to reside at a third host, and

(4) It permits multiple recipients.

The breakdown is the following:

COMMANDS THAT MUST BE IMPLEMENTED

(Author and Title could be treated as NOPs)

To enter the Mail subsystem:

MAIL <CA>

To invoke the Delivery function:

DELIVER <CA>

To specify the text of the message:

FILE <CA>

LOCATION <fileaddr> <CA>

TEXT <string> <CA2>

To identify author(s), recipient(s), and title:

AUTHOR <individual> <CA>

RECIPIENT <individual> <CA>

TITLE <title> <CA>

To exit the function or subsystem:

ABORT <CA>

EXIT <CA>

COMMANDS THAT CAN BE TREATED AS NOPS

(they can legally appear in the Delivery function)

ACCESS <individual> <CA>

ACCESSTYPES <accesstypes> <CA>

CATALOG <catalog> <CA>

CLERK <individual> <CA>

COMMENTS <comments> <CA2>

CREATIONDATE <datetime> <CA>

DELIVERYTYPE <deliverytype> <CA>

DISPOSITION <disposition> <CA>

GENERALDELIVERY <CA>

GREETING <greeting> <CA>

ID <id> <CA>

REFERENCE SERIAL <serialnumber> <CA>

SERIAL <serialnumber> <CA>

SIGNATURE <signature> <CA>

COMMANDS THAT NEEDN'T BE RECOGNIZED

(they cannot legally appear in the Delivery function)

Commands that invoke unsupported functions:

DISTRIBUTE <CA>

FORWARD <CA>

RECORD <CA>

RETRIEVE <CA>

UPDATE <CA>

VERIFY <CA>

Miscellaneous parameter specification commands:

ACKCONDITION <ackcondition> <CA>

ACKTYPE <acktype> <CA>

CITATIONTEMPLATE <citationtemp> <CA>

CUTOFF <interval> <CA>

FORWARDEE <individual> <CA>	7a2c3b5
MONITOR <individual> <CA>	7a2c3b6
PATHNAME <pathname> <CA>	7a2c3b7
REPORTINTERVAL <interval> <CA>	7a2c3b8
REQUESTOR <individual> <CA>	7a2c3b9
UPDATETYPE <updatetype> <CA>	7a2c3b10

CA AND CA2 NOT EXPLAINED SCON ENOUGH 7b

References:	7b1
(DHC JBP RFC 539 -- 17644,3a:gy)	7b1a
Discussion:	7b2
Agreed.	7b2a

CHANGE 'INTERRUPT' TO 'URGENT' OR 'PRIORITY' 7c

References:	7c1
(DHC JBP RFC 539 -- 17644,3e:gy)	7c1a
Discussion:	7c2
Agreed.	7c2a
How about 'URGENT'.	7c2b

CARRY STATIC/DYNAMIC ATTRIBUTE DISTINCTION INTO FORMAL SYNTAX 7d

References:	7d1
(DHC JBP RFC 539 -- 17644,3i:gy)	7d1a
Discussion:	7d2
Agreed.	7d2a

CRYPTIC DEFAULT DESCRIPTIONS 7e

References:	7e1
(DHC JBP RFC 539 -- 17644,3k:gy)	7e1a
Discussion:	7e2
Agreed.	7e2a

